

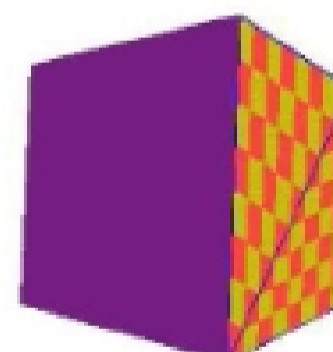
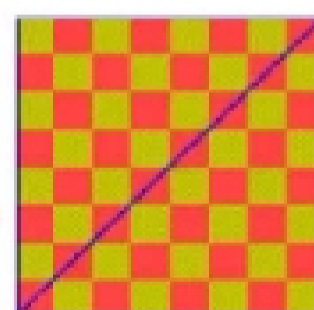
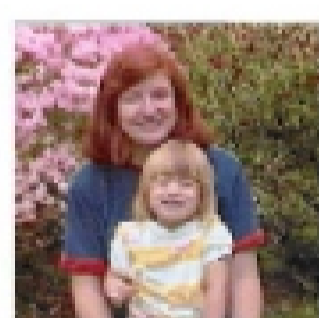
## Pipelines and Texture Mapping



- **Texture mapping is done at rasterization**
  - in canonical screen space
- **When describing a scene, you assume that texture interpolation will be done in world space**
- **Which property of perspective projection means that the “wrong thing” will happen if we apply our simple interpolations from the previous slide?**
- ***Perspective correct texture mapping* does the right thing, but at a cost**
- **Is it a problem with orthographic viewing?**



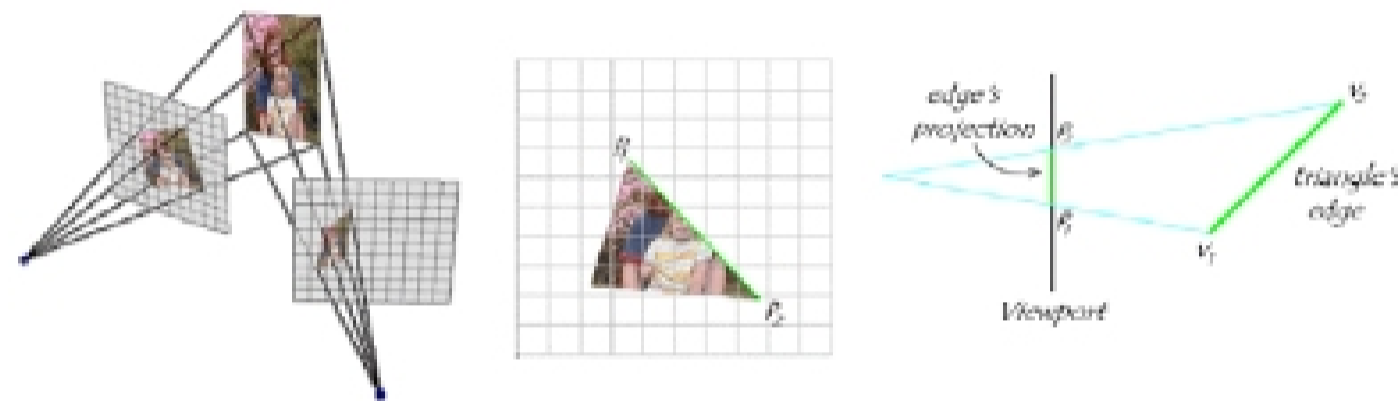
## Linear Interpolation of Textures



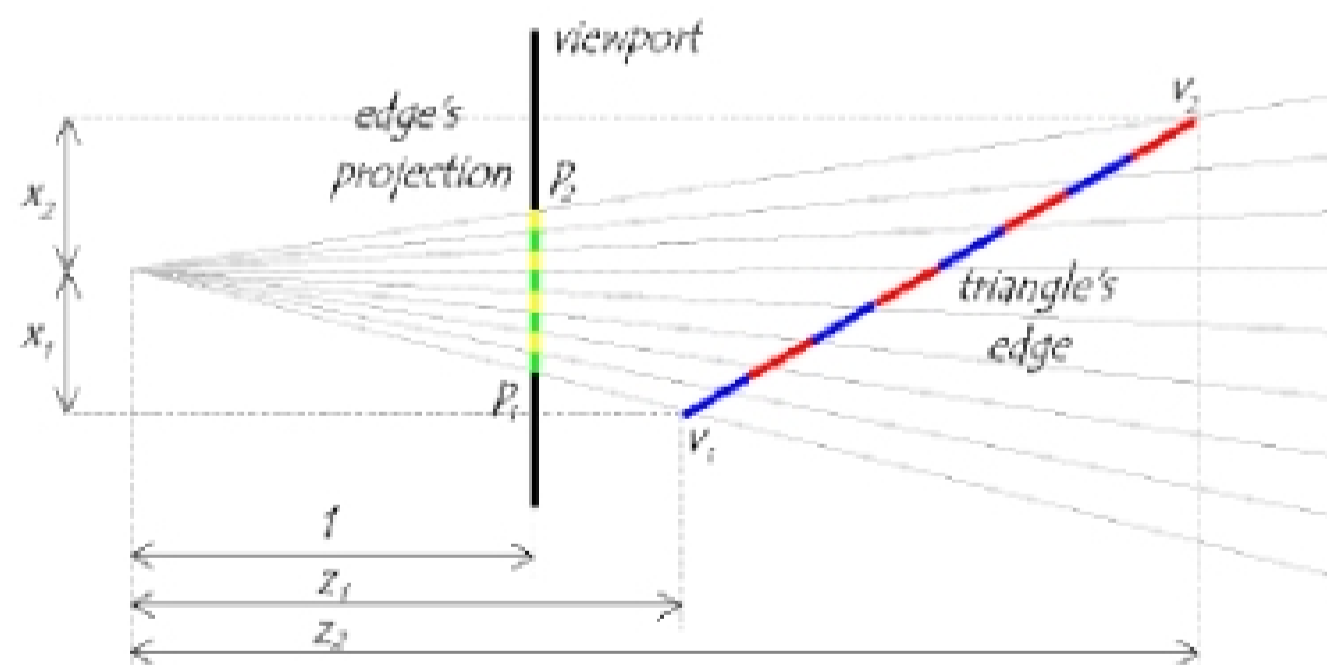
## Texture Index Interpolation

Interpolating texture indices is not as simple as the linear interpolation of colors that we discussed when rasterizing triangles. Let's look at an example.

First, let's consider one edge from a given triangle. This edge and its projection onto our viewport lie in a single common plane. For the moment, let's look only at that plane, which is illustrated below:



## Texture Interpolation Problem



*Notice that uniform steps on the image plane do not correspond to uniform steps along the edge.*



## Perspective Correction Hint

- **Texture coordinate and color interpolation**
  - either linearly in screen space
  - or using depth/perspective values (slower)
- **Noticeable for polygons “on edge”**

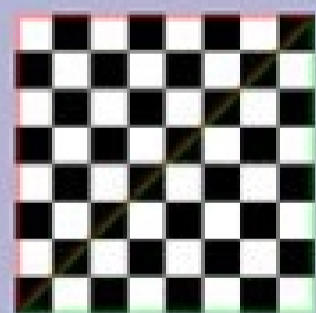
```
glHint( GL_PERSPECTIVE_CORRECTION_HINT, hint )
```

where *hint* is one of

- *GL\_DONT\_CARE*
- *GL\_NICEST*
- *GL\_FASTEST*



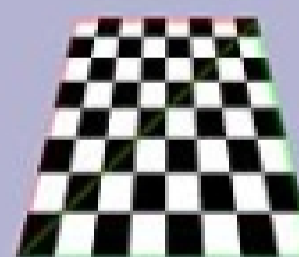
## Perspective Correct Texturing



Flat



Affine



Correct

